RECEIVED
CENTRAL FAX CENTER
JUL 24 2006

REMARKS

Claims 1-17 have been rejected under 35 U.S.C. §103(a) as being unpatentable over the admitted prior art of U.S. Patent Application Publication No.: 2002/0087590 to Bacon et al. ("Bacon") in view of U.S. Patent No. 6,338,159 Alexander, III et al. ("Alexander").

Claims 1-17 remain pending.

Rejection of claims 1-17 under 35 U.S.C. §103(a)

With respect to independent claim 1, the Office Action states that the combination of Bacon and Alexander teaches all of Applicant's recited elements.

Claim 1 has been amended to point out more clearly what Applicant regards as the invention. Specifically, claim 1 has been amended to add the limitation wherein each of the plurality of objects can be pointed to by one or more other objects and wherein each of the plurality of objects can be part of, and have different positions in, different reference chains such that the reference count and depth value change as the other objects that point to each of the plurality of objects change Support for the claim amendment can be found, at least, in paragraphs 0030-0033 and Figs. 2A-2E of the specification.

Bacon teaches techniques that allow concurrent collection of cyclic garbage on reference counting systems. Candidate objects are found that may be part of cyclic garbage. Each candidate object has a reference count. Two tests are performed to determine if concurrent operations have affected the reference counts of the candidate objects. If concurrent operations have not affected the reference counts, the candidate objects are collected as garbage. Additionally, during garbage collection, the decrements to reference counts are delayed so that increments occur before decrements and so that decrements are held a predetermined time before

being applied. This prevents decrementing a reference count and collecting a cycle as garbage right before a reference is added to an object in the cycle.

As admitted by the Examiner, Bacon does not teach or suggest maintaining a depth value based on a distance from a global object data, and identifying, based on the associated depth value, which of the plurality of objects are to be processed to determine whether or not they are garbage.

Alexander teaches a system and method for representing program event trace information in a way, which supports a variety of queries regarding system performance. The tracing and reduction of the system and method may be dynamic, in which case information is obtained and added to the trace representation in real-time. Alternately, the tracing and reduction may be static, in which case a trace text file or binary file is obtained from a trace buffer, and the reduction takes place using the trace file as input. The trace information, whether obtained statically or dynamically, is represented as a tree of events. The system and method may be used to present many types of trace information in a compact manner, which supports performance queries. For example, the tree structure may reflect the call stacks observed during a program's execution, and statistics regarding the memory allocated/deallocated in the various routines and call stacks may be stored at each node of the tree. The tree structure may be used to store performance information regarding Java bytecodes executed, memory allocated, or other types of performance information.

In contrast, Applicants' invention recites a garbage collecting method for a memory resource in a computer system. The method includes, for each of a plurality of objects in the memory resource, maintaining a reference count based on a number of objects pointing thereto, and maintaining a depth value based on a distance from a global data object in a reference chain.

The method further includes identifying, based on the associated reference count and depth value, which of the plurality of objects are processed to determine whether or not they are garbage, wherein each of the plurality of objects can be pointed to by one or more other objects and wherein each of the plurality of objects can be part of, and have different positions in, different reference chains such that the reference count and depth value change as the other objects that point to each of the plurality of objects change.

The Examiner cites col. 5, lines 24-27 and Fig. 5 of Alexander as allegedly teaching maintaining a depth value based on a distance from a global object data, and identifying, based on the associated depth value, which of the plurality of objects are to be processed to determine whether or not they are garbage.

The purpose of the system and method taught by Alexander is to provide trace information to a developer or systems manager in a compact and efficient manner, and to provide trace information in such a way to facilitate understanding of the system's operation. Fig. 3 of Alexander shows a portion of a trace sequence, along with the state of the call stack after each trace event. The trace text file is created based on information contained in a trace buffer (col. 4, lines 19-31. Fig. 4, which was cited by the Examiner, is an event tree, which reflects call stacks observed during system execution. At each node in the tree, several statistics are recorded. The particular statistics shown include the number of distinct times the call stack is produced, the sum of the time spent in the call stack, the total time spent in the call stack plus the time in those call stacks invoked from this call stack, and the number of instances of this routine above this instance (col. 4, lines 37-47).

The tree structure shown in Fig. 4 of Alexander depicts one way in which data maybe be pictorially presented to a user. The same data may also be presented to a user in tabular form as

shown in Fig. 5 of Alexander (col. 5, lines 18-21).

The cited passages at col. 5, lines 24-27 only teach that Table 5 includes columns of data for Level, RL, Calls, Base, Cum, and Indent, and that the Level is the tree level (counting from the root as level 0) of the node. Each node can be traced back to only one root. In other words, no node is a child of two separate roots. Further, the Level (level of the node in the tree) is not used to determine anything.

In contrast, Applicant's invention teaches the depth value of an object as the distance of the object from a global data object (the object's position in a reference chain) (See paragraph 0030 of the specification). The object can also be referred to by multiple data objects (See Fig. 2B), and/or be part of, and have different positions in, different reference chains such that the reference count and depth value change as the other objects that point to the object change (See Fig. 2B), unlike a node level in a tree (as taught by Alexander). A node level in a tree, as taught by Alexander, can only be pointed to by one parent object, and cannot change its position in the tree.

Further, the object recited in Applicant's claims can also be referred to by a subsequent object in the same reference chain (See Fig. 3F). The depth level of Applicant's invention can have a value of 1 to 3, or be uninitialized. If an object in a reference chain has a depth value of 3, a subsequent object in the reference chain will have a depth value of 1, not 4 (See Fig. 2F and paragraph 0034), unlike the node level taught by Alexander.

An object is identified for processing by a garbage collector to determine with finality whether the object is garbage when it is determined that a second object previously pointing to the first object is no longer pointing to the first object, and the depth value of the second object is

one less than the depth value of the first object. The node level taught by Alexander is not used in any such determination.

It is clear that the level of the node taught by Alexander is not at all similar to the depth value recited in Applicant's independent claim 1. Therefore, Alexander does not teach or suggest maintaining a depth value based on a distance from a global object data, and identifying, based on the associated depth value, which of the plurality of objects are to be processed to determine whether or not they are garbage, wherein each of the plurality of objects can be pointed to by one or more other objects and wherein each of the plurality of objects can be part of, and have different positions in, different reference chains such that the reference count and depth value change as the other objects that point to each of the plurality of objects change.

In view of the foregoing, it is respectfully submitted that Bacon and Alexander, whether taken alone or in combination, do not teach or suggest the subject matter recited in Applicant's independent claim 1. Specifically, the references do not teach or suggest a garbage collecting method for a memory resource in a computer system including, for each of a plurality of objects in the memory resource, maintaining a reference count based on a number of objects pointing thereto, and maintaining a depth value based on a distance from a global data object in a reference chain, and identifying, based on the associated reference count and depth value, which of the plurality of objects are processed to determine whether or not they are garbage, wherein each of the plurality of objects can be pointed to by one or more other objects and wherein each of the plurality of objects can be part of, and have different positions in, different reference chains such that the reference count and depth value change as the other objects that point to each of the plurality of objects change.

Claims 2-15, which depend directly or indirectly from the independent claim 1, incorporate all of the limitations of claim 1 and are therefore patentably distinct over Bacon and Alexander for at least those reasons provided for independent claim 1.

Independent claims 16 and 17 have been amended to recite limitations similar to those recited in amended independent claim 1, and are therefore patentably distinct over Bacon and Alexander for at least those reasons provided for independent claim 1.

Conclusion

In view of the foregoing, Applicant respectfully requests reconsideration, withdrawal of all rejections, and allowance of all pending claims in due course.

Respectfully submitted,

Steven Fischman
Registration No. 34,594

SCULLY, SCOTT, MURPHY & PRESSER
400 Garden City Plaza
Suite 300
Garden City, New York 11530
(516) 742-4343

SF:BMM:me

G:\IBM\1545\16472\AMEND\16472.am2.doc